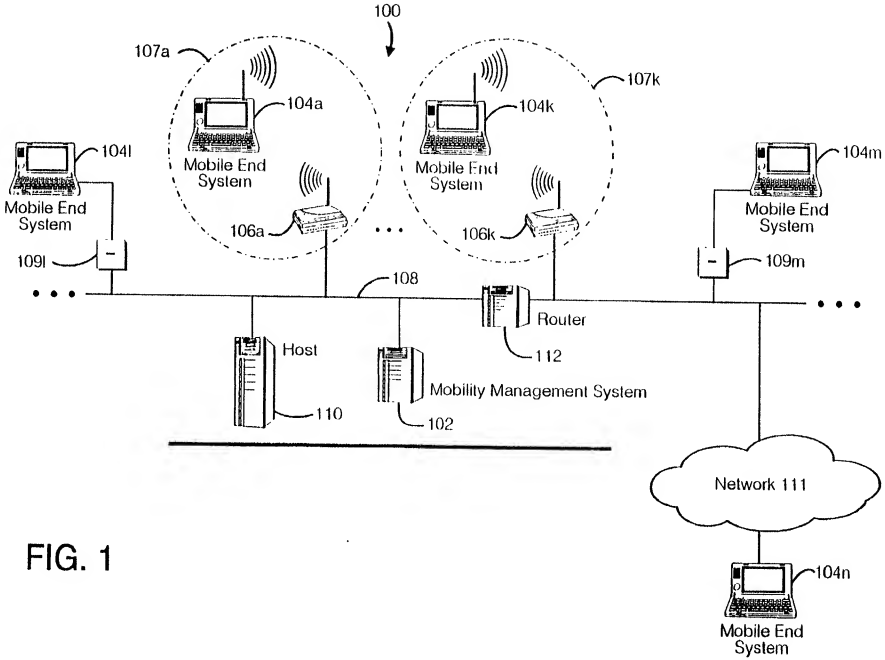


FIG. 1

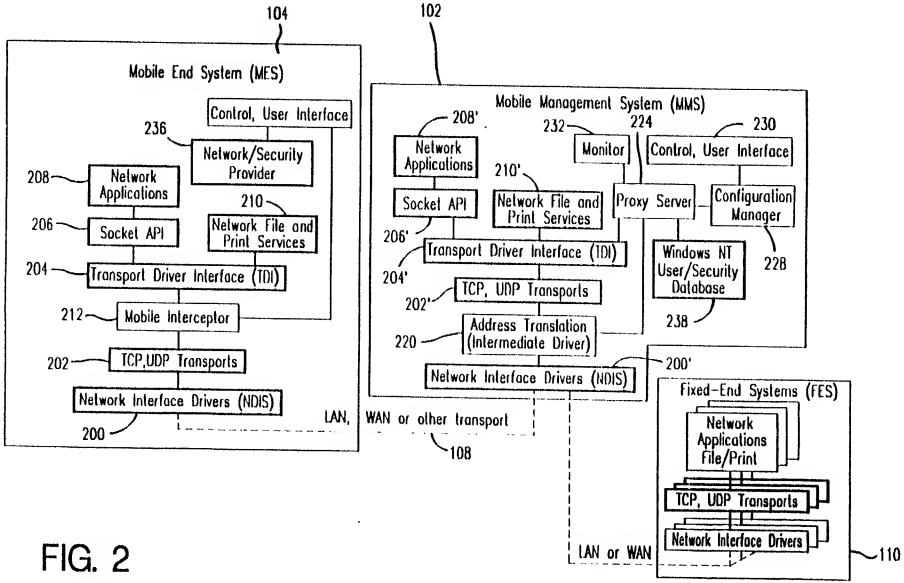

104
Mobile End System (MES)
Control, User Interface
236
Network/Security Provider
208
Network Applications
210
Network File and Print Services
206
Socket API
204
Transport Driver Interface (TDI)
212
Mobile Interceptor
202
TCP,UDP Transports
Network Interface Drivers (NDIS)
200
LAN,
102
Mobile Management System (MMS)
208'
Network Applications
232
Monitor
224
Control, User Interface
230
210'
Socket API
Network File and Print Services
Proxy Server
Configuration Manager
206'
Transport Driver Interface (TDI)
Windows NT User/Security Database
228
204'
TCP, UDP Transports
202'
238
220
Address Translation (Intermediate Driver)
200'
Network Interface Drivers (NDIS)
WAN or other transport
108
Fixed-End Systems (FES)
Network Applications File/Print
TCP, UDP Transports
Network Interface Drivers
LAN or WAN
110

FIG. 2

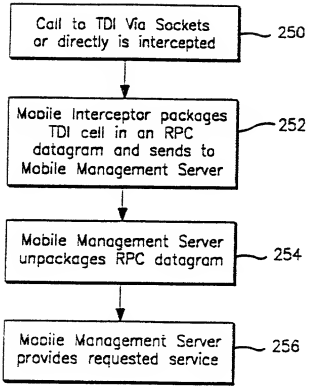

Call to TDI Via Sockets or directly is intercepted
250
Mobile Interceptor packages TDI call in an RPC datagram and sends to Mobile Management Server
252
Mobile Management Server unpackages RPC datagram
254
Mobile Management Server provides requested service
256

FIG. 2A

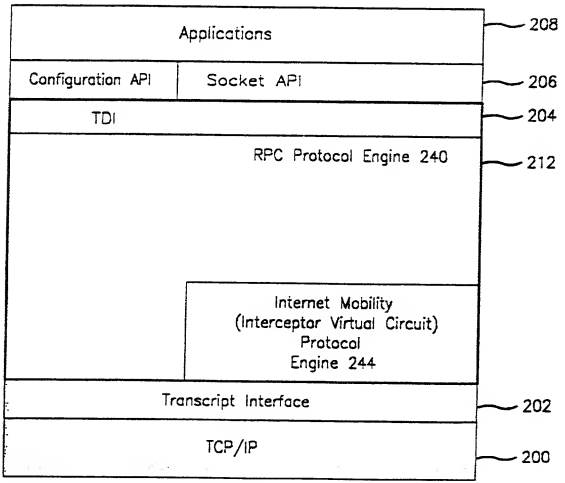

Applications
208
Configuration API
Socket API
206
TDI
204
RPC Protocol Engine 240
212
Internet Mobility (Interceptor Virtual Circuit) Protocol Engine 244
Transcript Interface
202
TCP/IP
200

FIG. 3

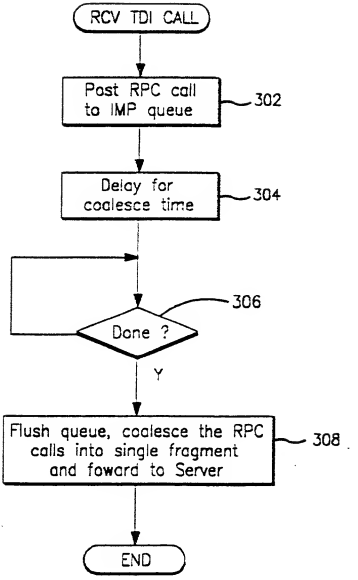

RCV TDI CALL
Post RPC call to IMP queue
302
Delay for coalesce time
304
306
Done ?
Y
Flush queue, coalesce the RPC calls into single fragment and foward to Server
308
END

FIG. 3A

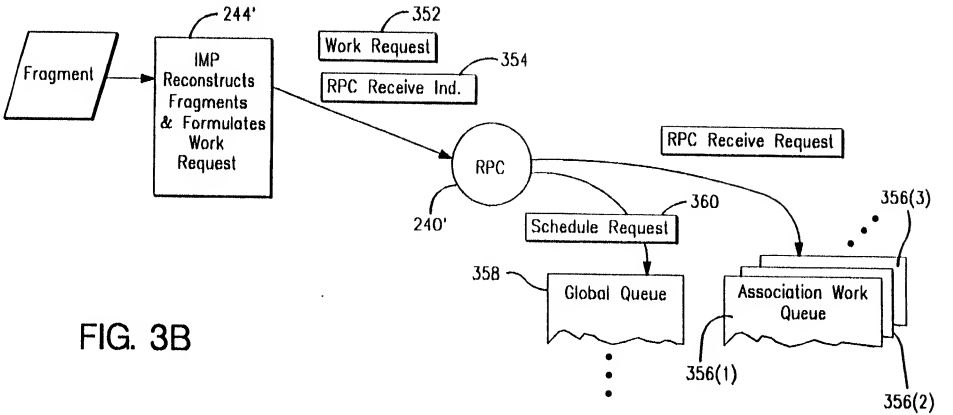


FIG. 3B

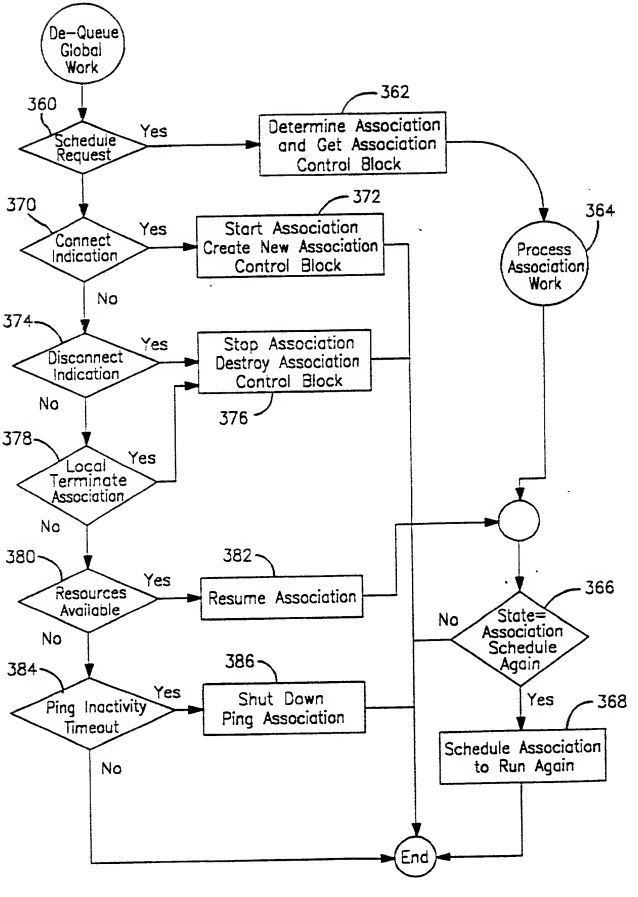

De-Queue Global Work
360
Schedule Request
Yes
362
Determine Association and Get Association Control Block
364
Process Association Work
370
Connect Indication
Yes
372
Start Association Create New Association Control Block
No
374
Disconnect Indication
Yes
Stop Association Destroy Association Control Block
376
No
378
Local Terminate Association
Yes
No
380
Resources Available
Yes
382
Resume Association
No
366
State= Association Schedule Again
No
384
Ping Inactivity Timeout
Yes
386
Shut Down Ping Association
Yes
368
Schedule Association to Run Again
No
End

FIG. 4

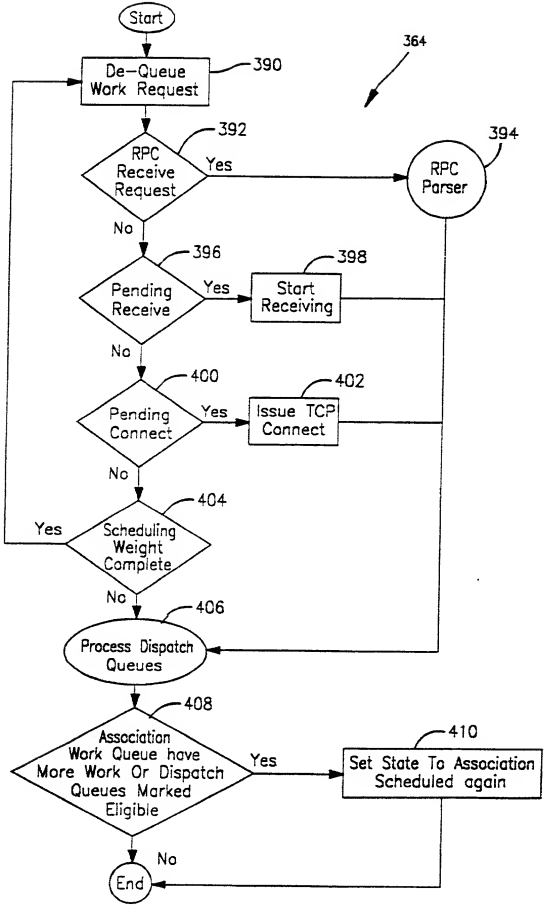


FIG. 5 Process Association Work

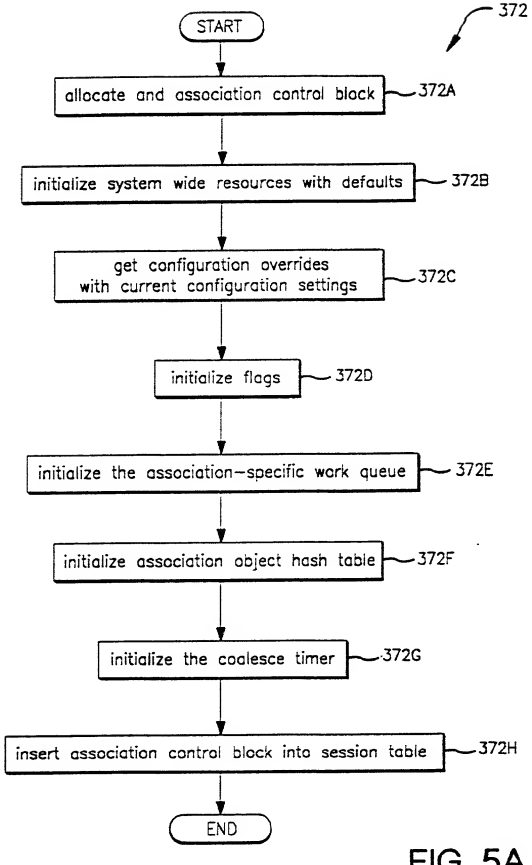

372
START
allocate and association control block
372A
initialize system wide resources with defaults
372B
get configuration overrides with current configuration settings
372C
initialize flags
372D
initialize the association–specific work queue
372E
initialize association object hash table
372F
initialize the coalesce timer
372G
insert association control block into session table
372H
END

FIG. 5A

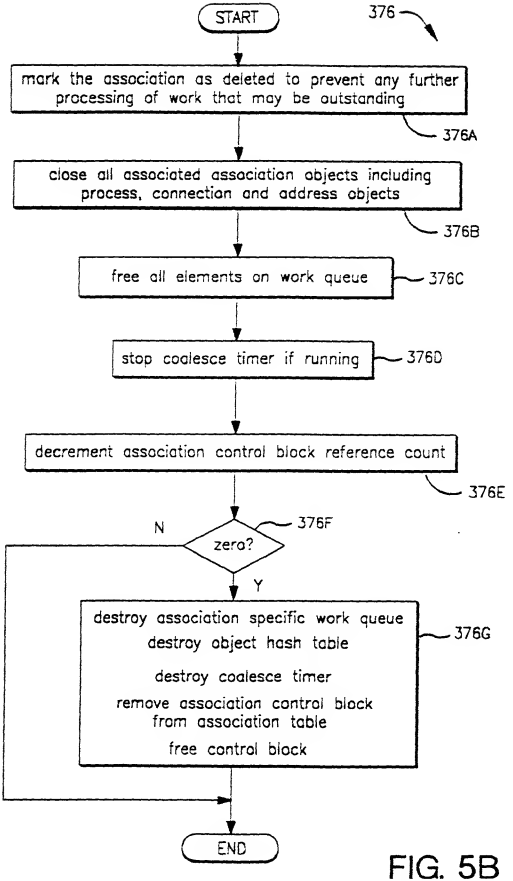

START
376
mark the association as deleted to prevent any further processing of work that may be outstanding
376A
close all associated association objects including process, connection and address objects
376B
free all elements on work queue
376C
stop coalesce timer if running
376D
decrement association control block reference count
376E
N
zero?
376F
Y
destroy association specific work queue
destroy object hash table
destroy coalesce timer
remove association control block from association table
free control block
376G
END

FIG. 5B

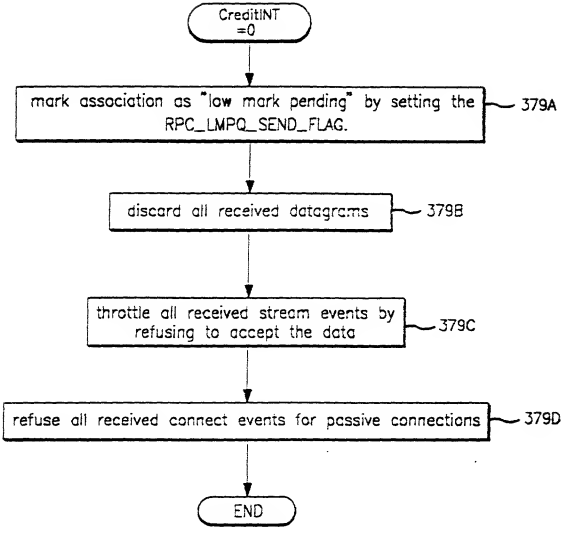

CreditINT =0
mark association as "low mark pending" by setting the RPC_LMPQ_SEND_FLAG.
379A
discard all received datagrams
379B
throttle all received stream events by refusing to accept the data
379C
refuse all received connect events for passive connections
379D
END

FIG. 5C

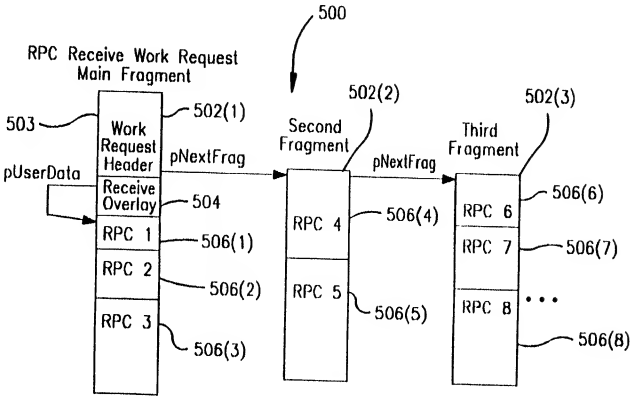


FIG. 6

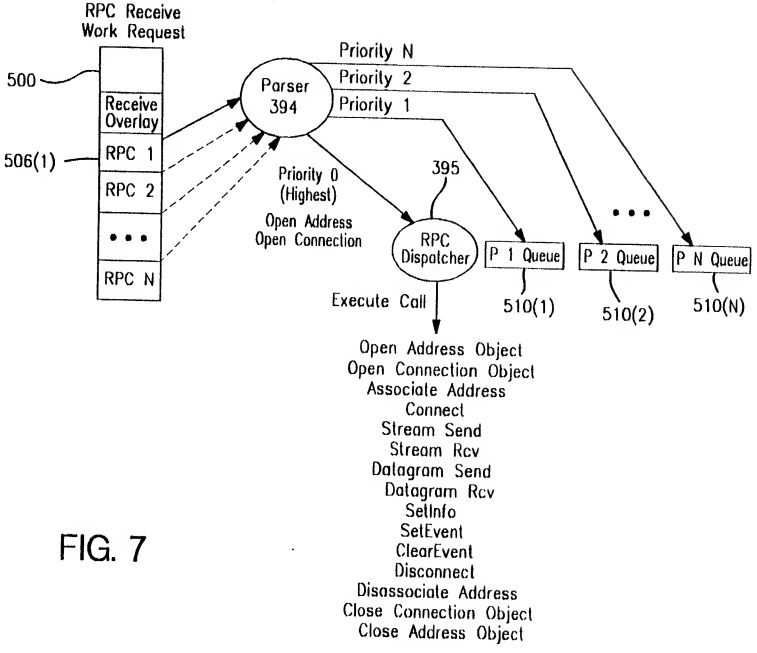

RPC Receive
Work Request
500
Receive
Overlay
506(1)
RPC 1
RPC 2
• • •
RPC N
Parser
394
Priority N
Priority 2
Priority 1
Priority 0
(Highest)
Open Address
Open Connection
395
RPC
Dispatcher
P 1 Queue
P 2 Queue
P N Queue
510(1)
510(2)
510(N)
Execute Call
Open Address Object
Open Connection Object
Associate Address
Connect
Stream Send
Stream Rcv
Datagram Send
Datagram Rcv
SetInfo
SetEvent
ClearEvent
Disconnect
Disassociate Address
Close Connection Object
Close Address Object

FIG. 7

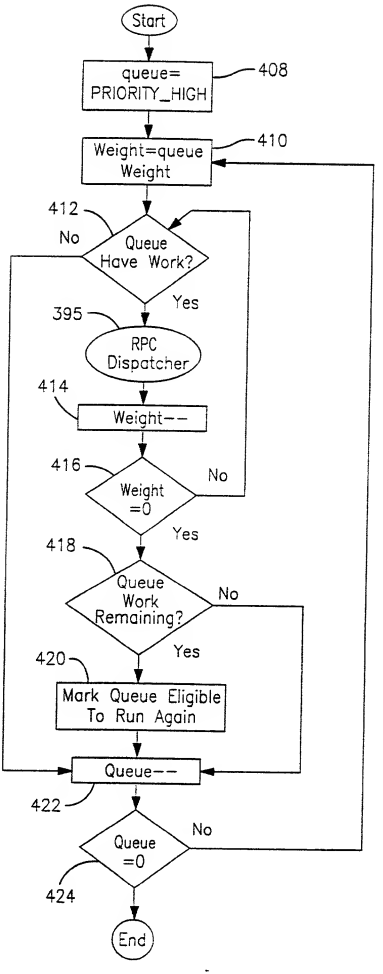


Fig. 8
Process Priority
Dispatch Queues

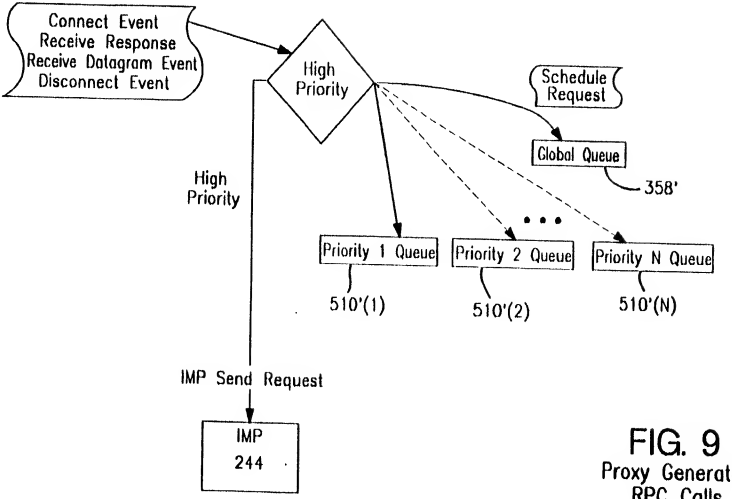
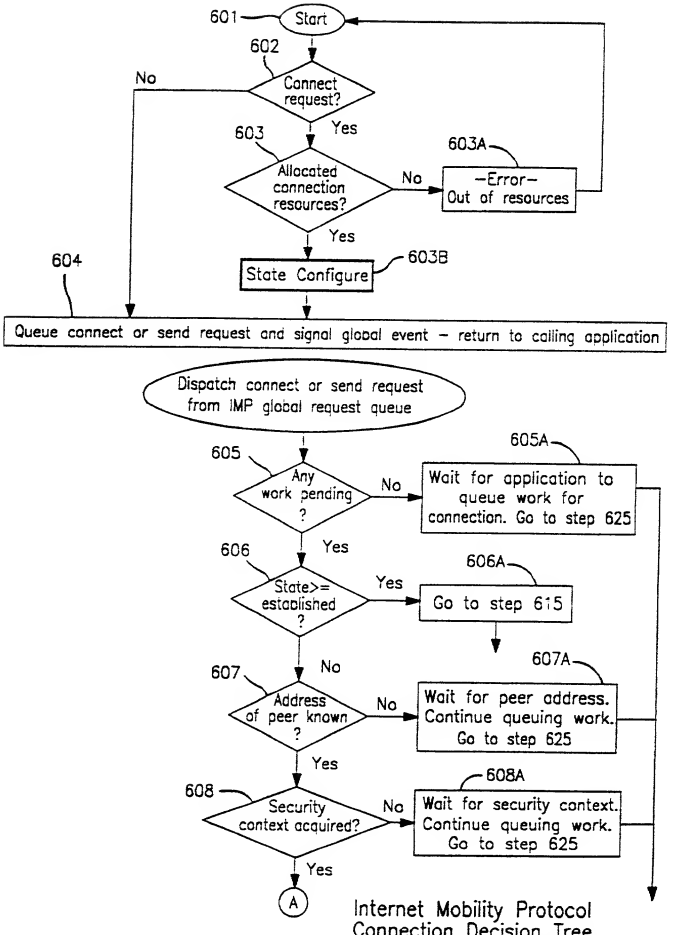


FIG. 9
Proxy Generated RPC Calls



Internet Mobility Protocol Connection Decision Tree.

FIG. 10A Connect and Send request logic

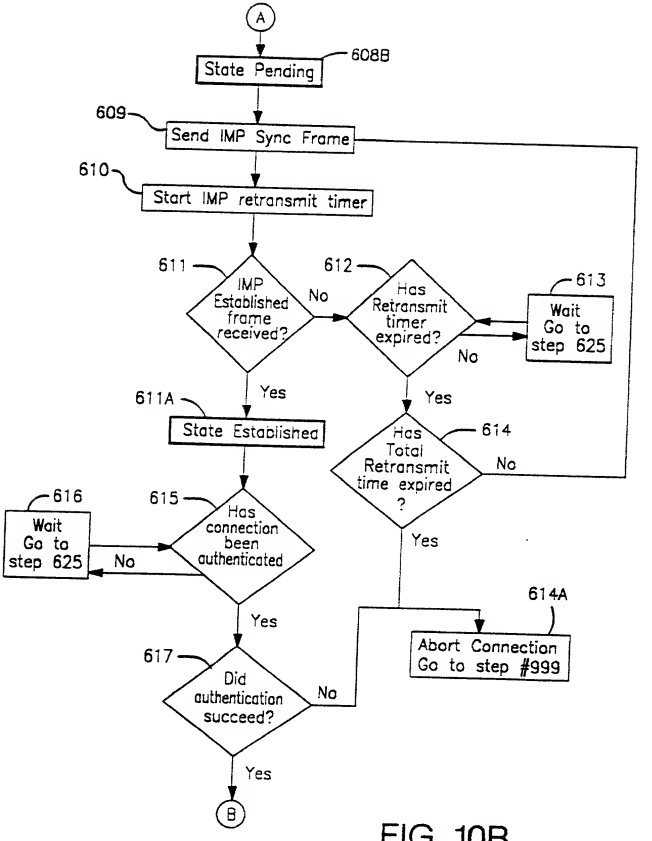

A
State Pending
608B
609
Send IMP Sync Frame
610
Start IMP retransmit timer
611
IMP Established frame received?
No
612
Has Retransmit timer expired?
613
Wait Go to step 625
No
Yes
611A
State Established
Yes
614
Has Total Retransmit time expired ?
No
616
Wait Go to step 625
615
Has connection been authenticated
No
Yes
614A
Abort Connection Go to step #999
617
Did authentication succeed?
No
Yes
B

FIG. 10B

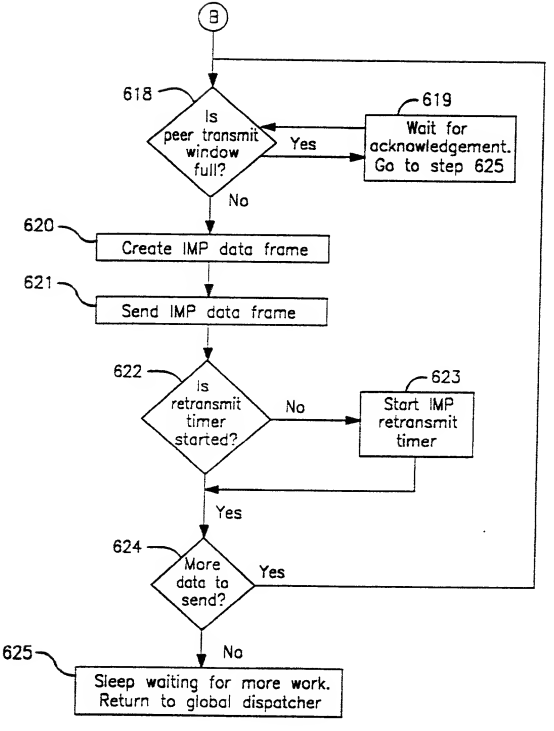

B
618
Is peer transmit window full?
Yes
619
Wait for acknowledgement. Go to step 625
No
620
Create IMP data frame
621
Send IMP data frame
622
Is retransmit timer started?
No
623
Start IMP retransmit timer
Yes
624
More data to send?
Yes
625
No
Sleep waiting for more work. Return to global dispatcher

FIG. 10C

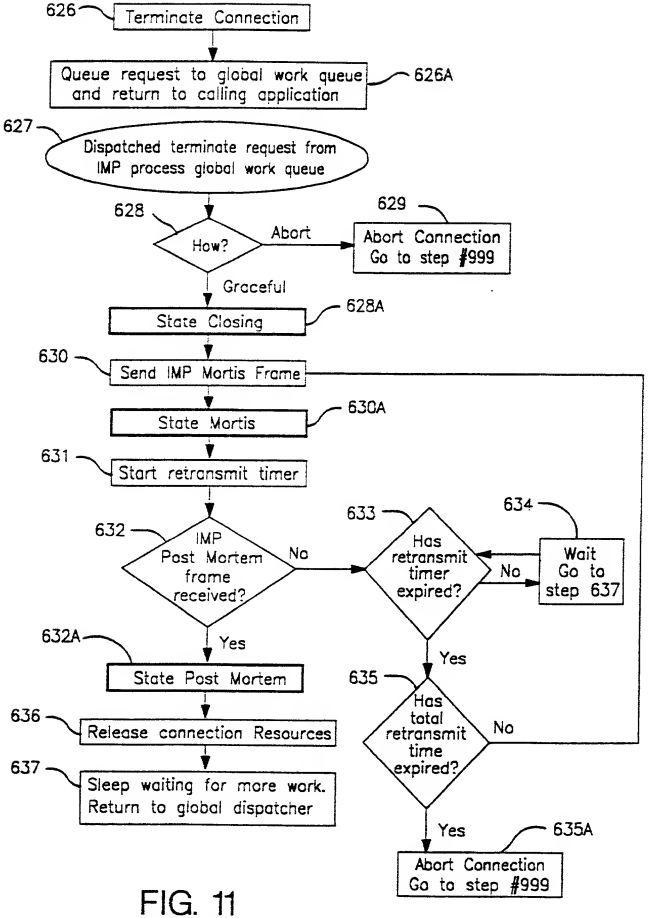


FIG. 11

Terminate Connection request logic

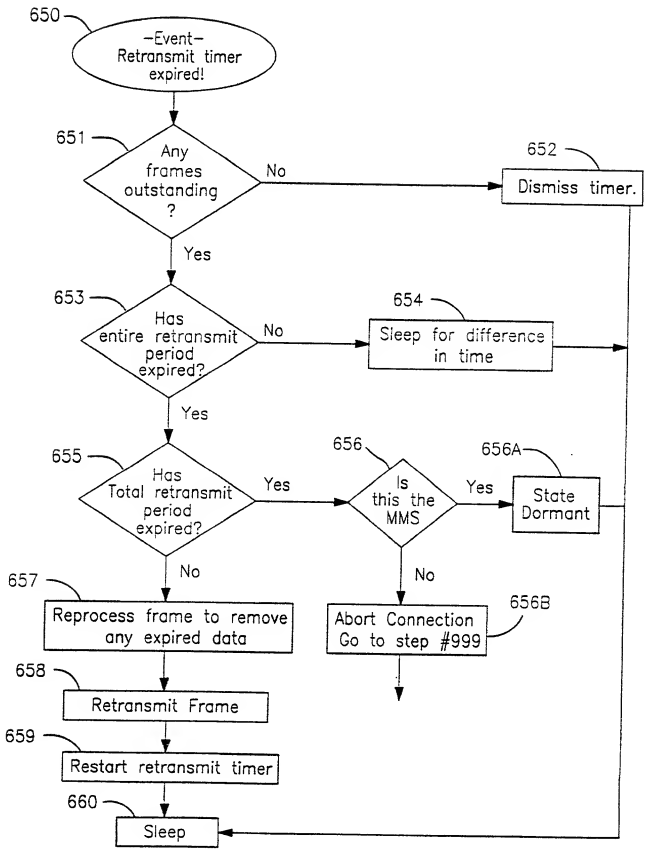


Fig. 12

Retransmit Event Logic

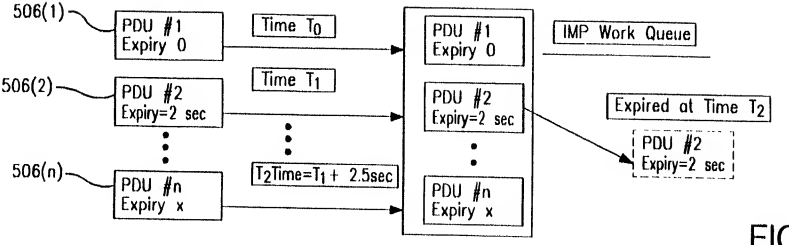


FIG. 12A

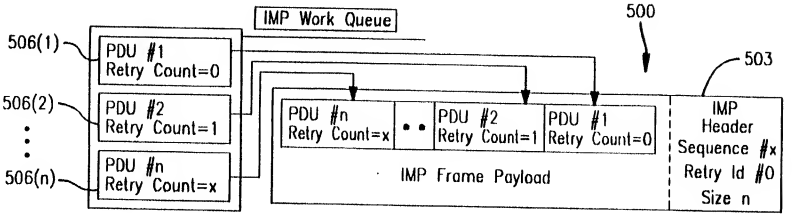


FIG. 12B

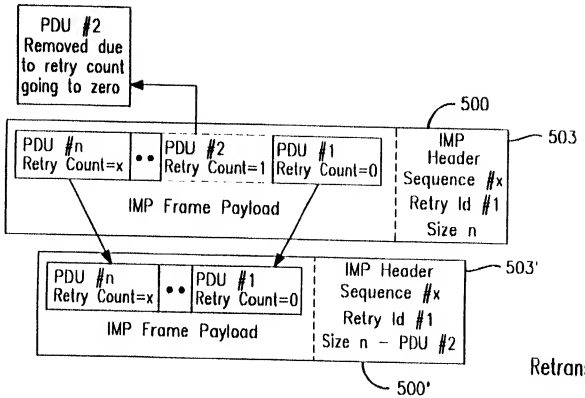


FIG. 12C

Retransmission of IMP Frame

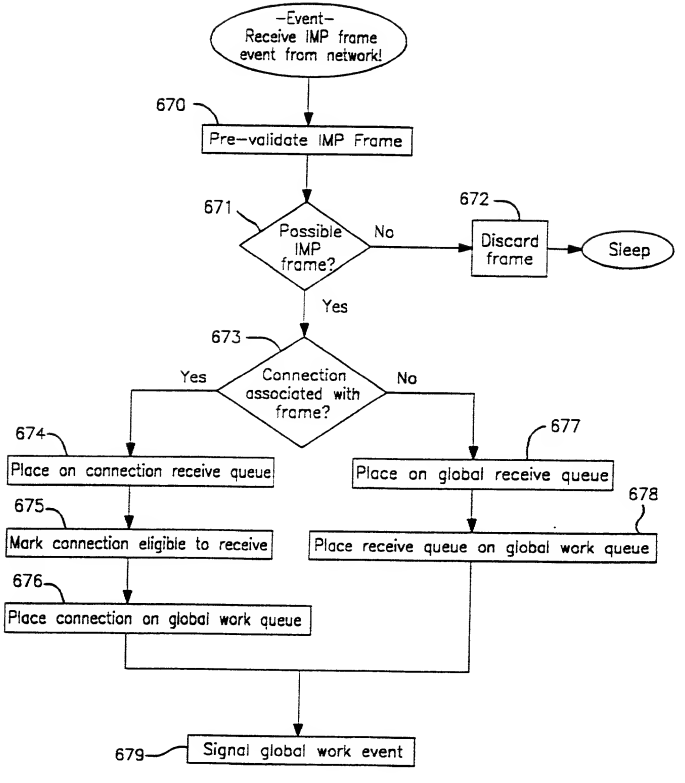


FIG. 13A

Receive Event Logic

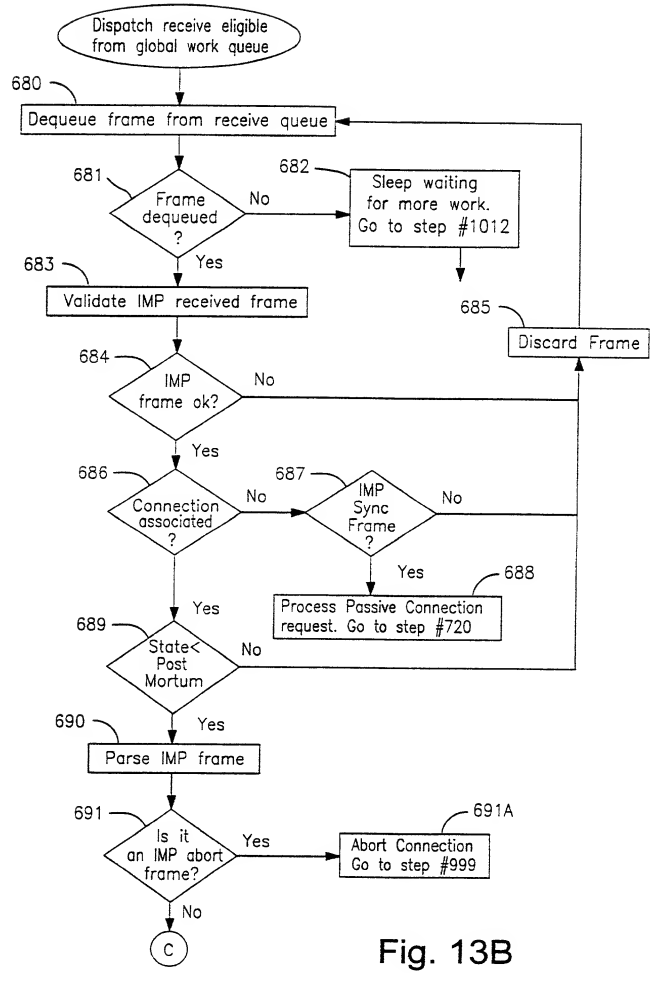

Dispatch receive eligible from global work queue
680
Dequeue frame from receive queue
681
Frame dequeued ?
No
682
Sleep waiting for more work. Go to step #1012
Yes
683
Validate IMP received frame
685
Discard Frame
684
IMP frame ok?
No
Yes
686
Connection associated ?
No
687
IMP Sync Frame ?
No
Yes
688
Process Passive Connection request. Go to step #720
Yes
689
State< Post Mortum
No
690
Yes
Parse IMP frame
691
Is it an IMP abort frame?
Yes
691A
Abort Connection Go to step #999
No
C

Fig. 13B

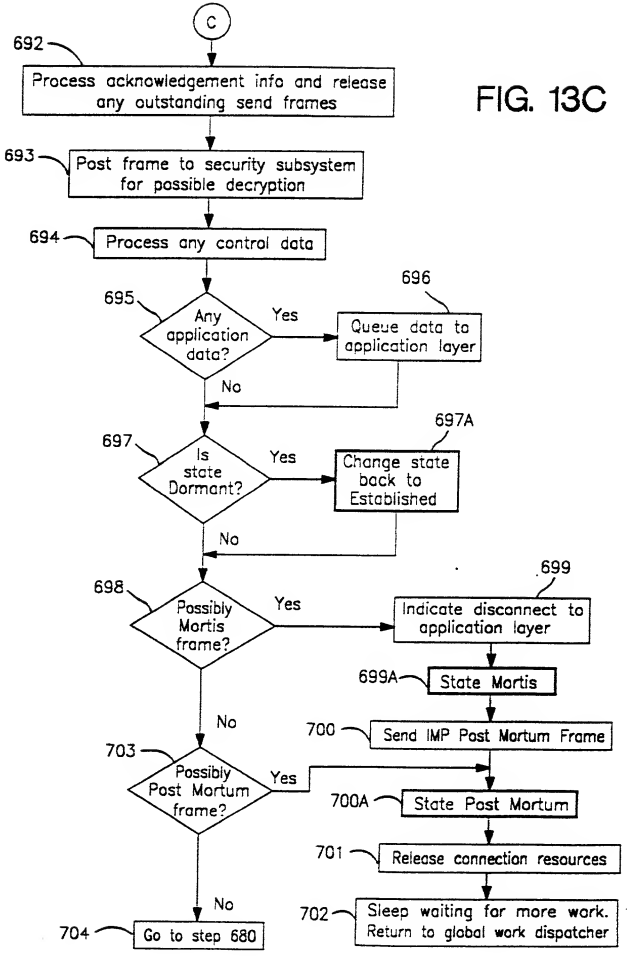

C
692
Process acknowledgement info and release any outstanding send frames
693
Post frame to security subsystem for possible decryption
694
Process any control data
695
Any application data?
Yes
696
Queue data to application layer
No
697
Is state Dormant?
Yes
697A
Change state back to Established
No
698
Possibly Mortis frame?
Yes
699
Indicate disconnect to application layer
699A
State Mortis
No
700
Send IMP Post Mortum Frame
703
Possibly Post Mortum frame?
Yes
700A
State Post Mortum
701
Release connection resources
No
702
Sleep waiting for more work. Return to global work dispatcher
704
Go to step 680

FIG. 13C

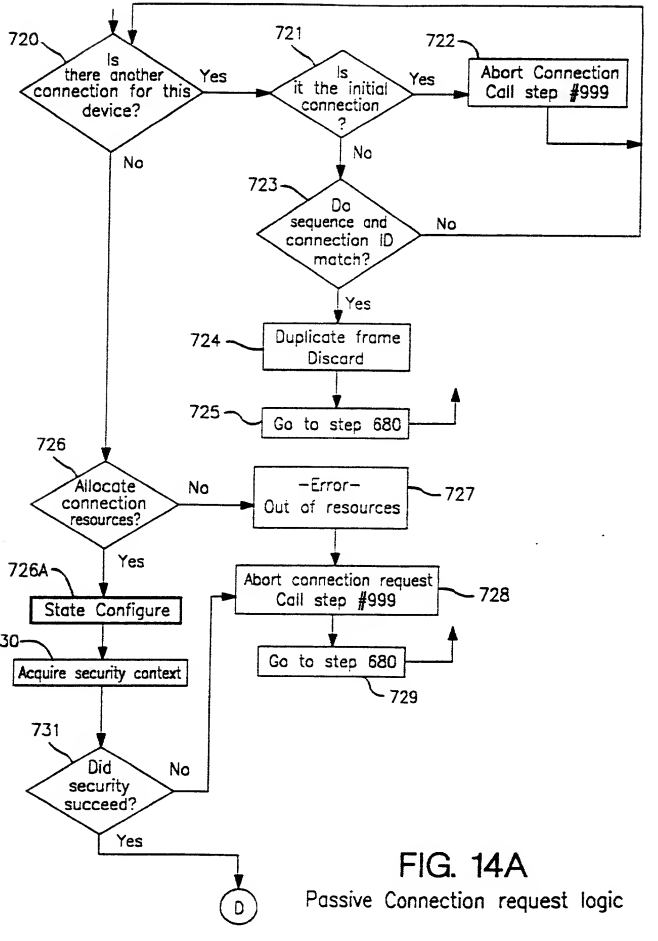


FIG. 14A

Passive Connection request logic

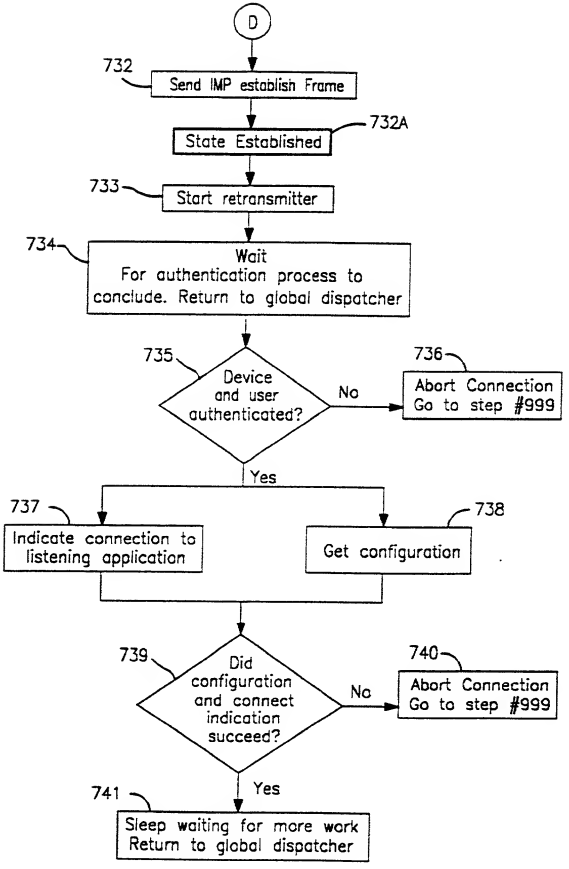

D
732
Send IMP establish Frame
732A
State Established
733
Start retransmitter
734
Wait
For authentication process to conclude. Return to global dispatcher
735
Device and user authenticated?
No
736
Abort Connection
Go to step #999
Yes
737
Indicate connection to listening application
738
Get configuration
739
Did configuration and connect indication succeed?
No
740
Abort Connection
Go to step #999
Yes
741
Sleep waiting for more work
Return to global dispatcher

FIG. 14B

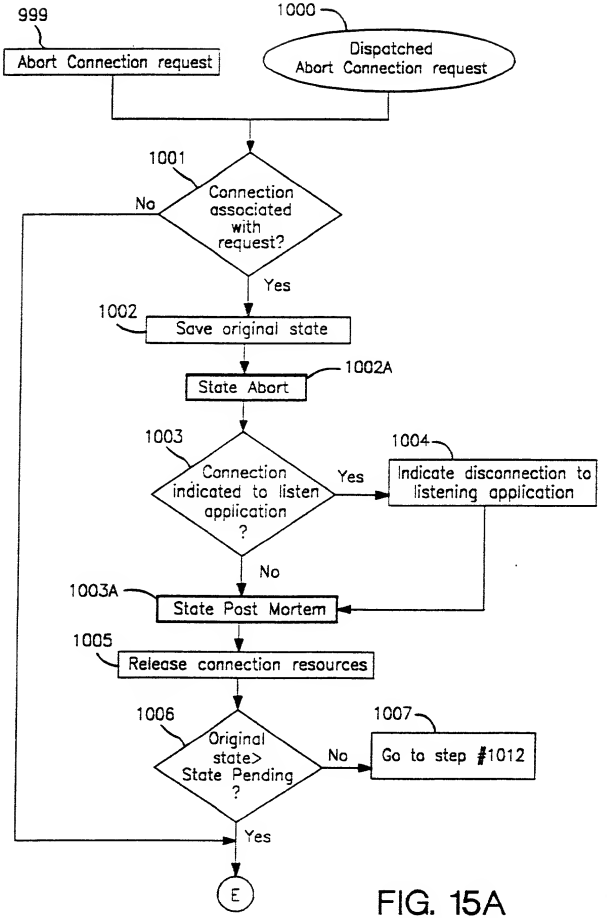


FIG. 15A

Abort Connection request logic

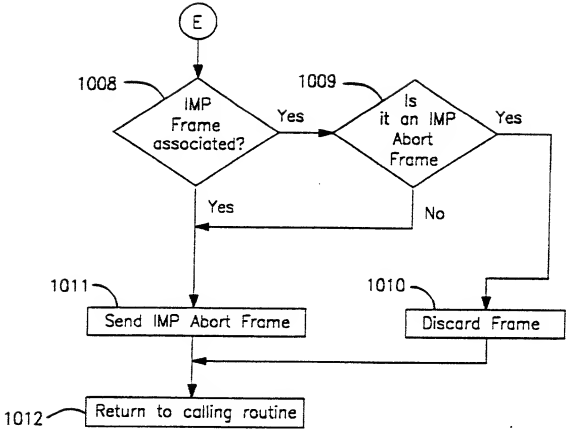

E
1008
IMP Frame associated?
Yes
1009
Is it an IMP Abort Frame
Yes
Yes
No
1011
Send IMP Abort Frame
1010
Discard Frame
1012
Return to calling routine

FIG. 15B

902

Socket:

servers	*linked list of server*
xid	*integer transaction ID number*
ping	*counter*
timeout	*time-out value that can be backed off*

904

Server:

next	*pointer to next server*
serverID	*IP Address of a DHCP server*
giaddr	*BOOTP Relay agent recently associated with this server*
ping	*c.f. socket ->ping*
new	*flag*

FIG. 16

DHCP Listener Data Structures

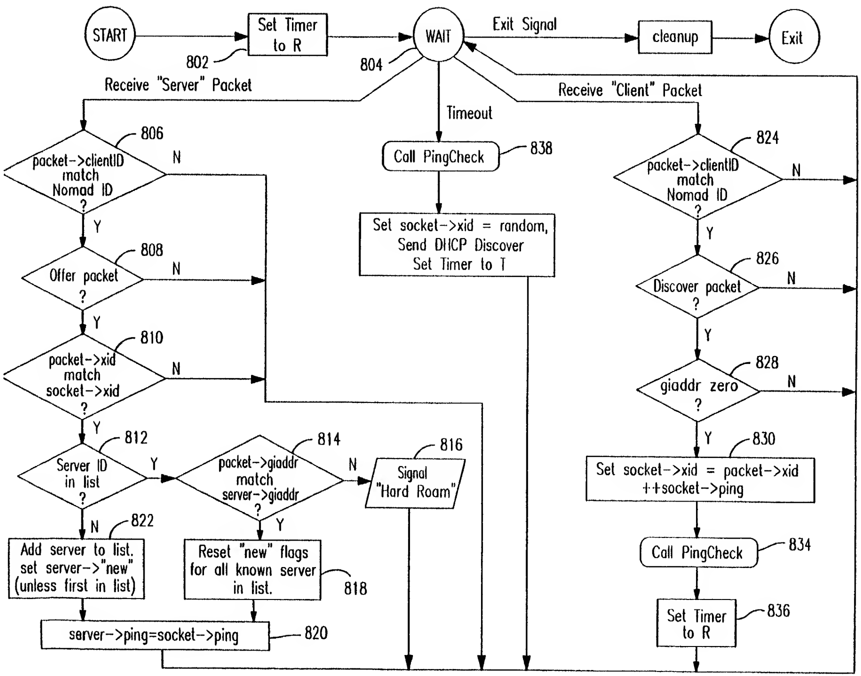
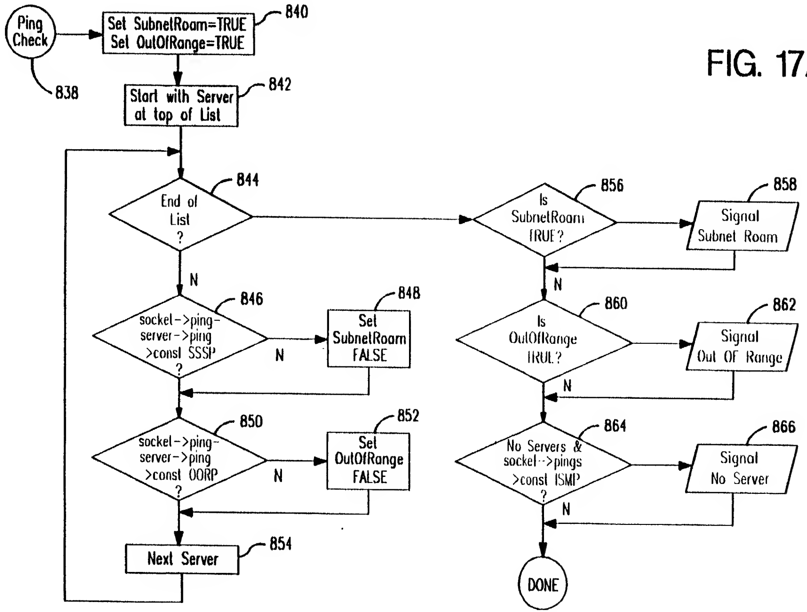

START
Set Timer to R
802
WAIT
804
Exit Signal
cleanup
Exit
Receive "Server" Packet
Receive "Client" Packet
Timeout
Call PingCheck
838
Set socket->xid = random, Send DHCP Discover Set Timer to T
packet->clientID match Nomad ID ?
806
N
Y
Offer packet ?
808
packet->xid match socket->xid ?
810
Server ID in list ?
812
packet->giaddr match server->giaddr ?
814
Signal "Hard Roam"
816
Add server to list. set server->"new" (unless first in list)
822
Reset "new" flags for all known server in list.
818
server->ping=socket->ping
820
packet->clientID match Nomad ID ?
824
Discover packet ?
826
giaddr zero ?
828
Set socket->xid = packet->xid ++socket->ping
830
Call PingCheck
834
Set Timer to R
836

Fig. 17

FIG. 17A



Ping Check
838
Set SubnetRoam=TRUE
Set OutOfRange=TRUE
840
Start with Server at top of List
842
End of List ?
844
N
sockel->ping-server->ping >const SSSP ?
846
N
Set SubnetRoam FALSE
848
sockel->ping-server->ping >const OORP ?
850
N
Set OutOfRange FALSE
852
Next Server
854
Is SubnetRoam TRUE?
856
Signal Subnet Roam
858
N
Is OutOfRange TRUE?
860
Signal Out OF Range
862
N
No Servers & sockel->pings >const ISMP ?
864
Signal No Server
866
N
DONE

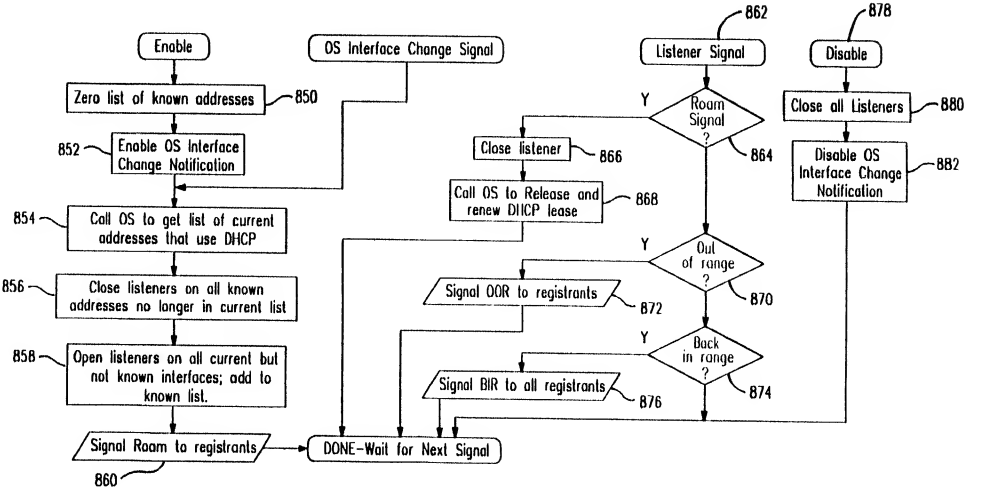


FIG. 18

ROAMING CONTROL CENTER–
Mobile End System

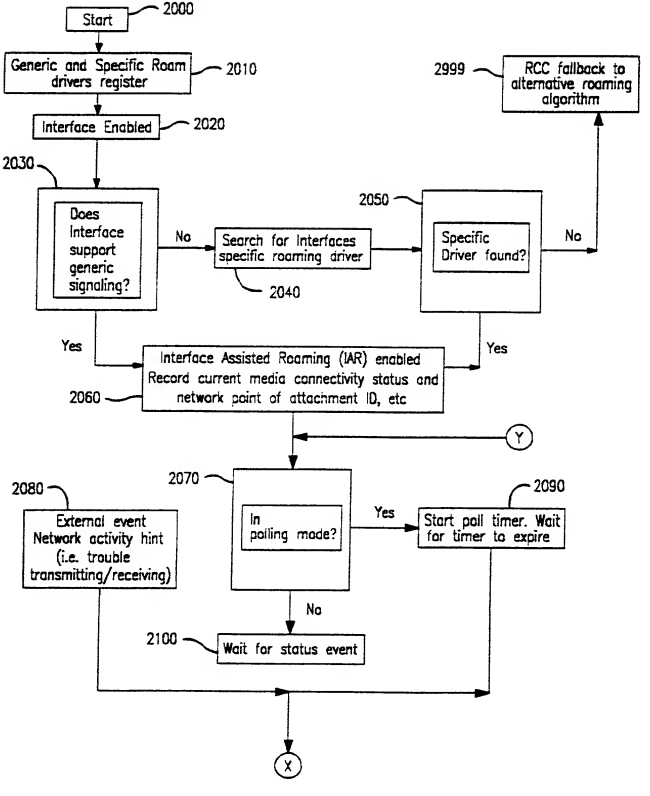
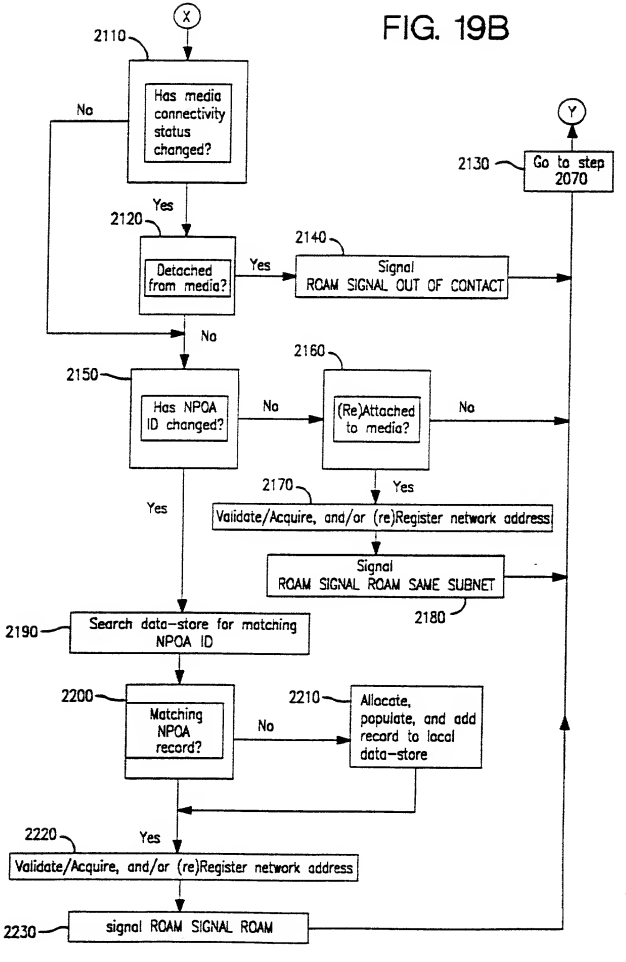


FIG. 19A

Interface Assisted Roaming
(IAR) Decision Tree

FIG. 19B


X
2110
Has media connectivity status changed?
No
Yes
2120
Detached from media?
Yes
2140
Signal ROAM SIGNAL OUT OF CONTACT
No
2150
Has NPOA ID changed?
No
2160
(Re)Attached to media?
No
Yes
2170
Yes
Validate/Acquire, and/or (re)Register network address
Signal ROAM SIGNAL ROAM SAME SUBNET
2180
Yes
2190
Search data-store for matching NPOA ID
2200
Matching NPOA record?
No
2210
Allocate, populate, and add record to local data-store
2220
Yes
Validate/Acquire, and/or (re)Register network address
2230
signal ROAM SIGNAL ROAM
Y
2130
Go to step 2070

Next Table Element	Previous Table Element	NPOA Unique Identifier	Network Level Address	Network Mask	Flags (i.e. Static Dynamic, etc.)	Timeout	Etc.

Next Table Element	Previous Table Element	NPOA Unique Identifier	Network Level Address	Network Mask	Flags (i.e. Static Dynamic, etc.)	Timeout	Etc.

•
•
•

Next Table Element	Previous Table Element	NPOA Unique Identifier	Network Level Address	Network Mask	Flags (i.e. Static Dynamic, etc.)	Timeout	Etc.

FIG. 20

Interface Assisted Roaming
Topology Node

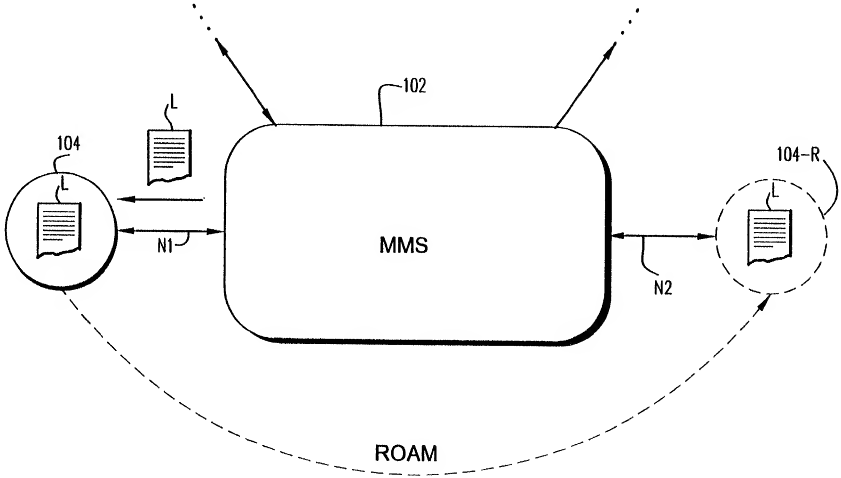


Fig.21

Disjoint network Roaming

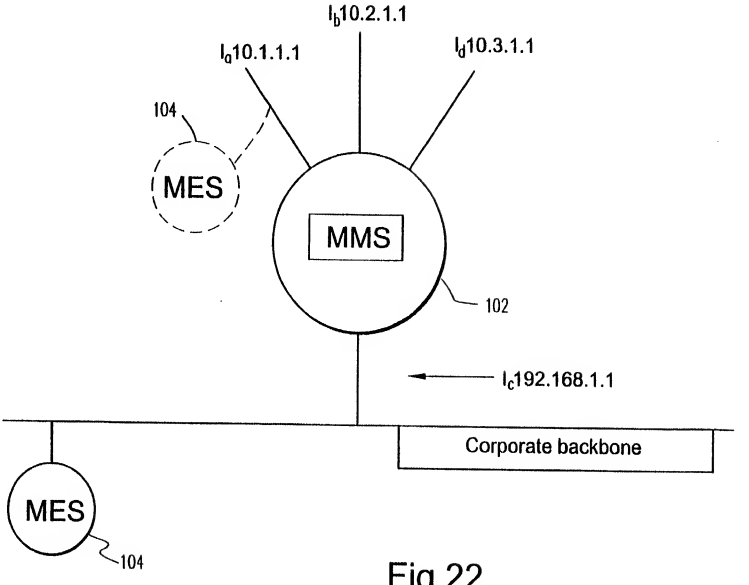
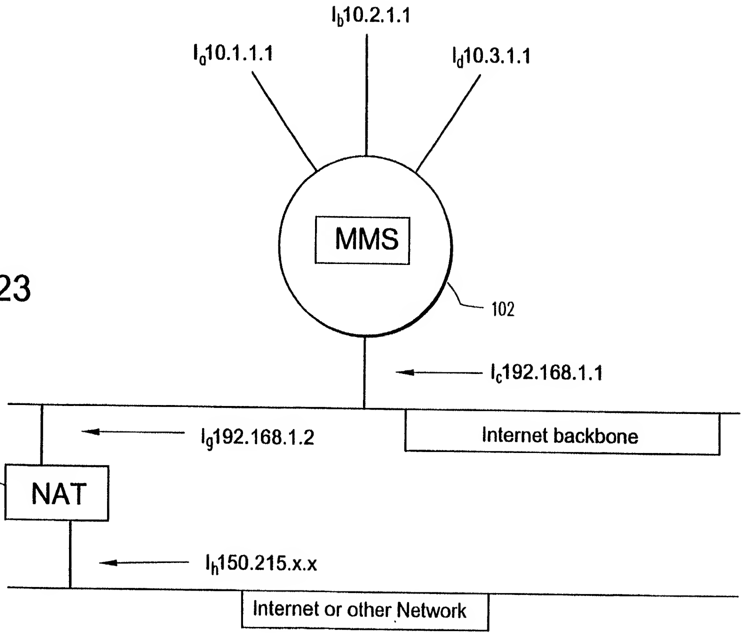


Fig.22

Example Secure Disjoint Coordination

Fig.23



I_a10.1.1.1
I_b10.2.1.1
I_d10.3.1.1
MMS
102
I_c192.168.1.1
I_g192.168.1.2
Internet backbone
2000
NAT
I_h150.215.x.x
Internet or other Network

FIG. 24

Example Policy Management Rules Table

TX/RX	Proxied	MES Source Port	MES Source Address	MES Dest Port	MES Dest Address	BPS (Available)	Process Name	Network	Location (GPS Coordinates)	Network Point of Attachment	User	Deny Request
T/R	Y	Any	Any	21	Any	< 100,000	Any	Any	Any	Any	US Patent Office	Y
T/R	Y	Any	Any	20	Any	< 100,000	Any	Any	Any	Any	US Patent Office	Y
T	N	5008	Any	5008	10.1.1.1		Any	Any	Any	Any	US Patent Office	N
R	N	5008	10.1.1.1	5008	Any		Any	Any	Any	Any	US Patent Office	N

Assumptions

1. Peer File Transfer Protocol control and data ports are 21 and 20
2. * indicates wildcard
3. MMS network address and port is 10.1.1.1: 5008
4. MES network port that frames from MMS is received on is 5008

In the example above all connections to destination ports 20 and 21 are denied or throttled if the available bandwidth is reduced to less then 100,000 bytes per second. In this example rules (rows) 3 and 4 only allow network traffic to flow to and from the MMS. All other network traffic that is not proxied is implicitly discarded. It should be appreciated that this table does not represent the full set of metrics that can be defined for policy management. Others variables such as monetary cost, location, network point of attachment, etc. can be added to the decision tree. Furthermore, the rules engine interpreting these entries can be distributed between the MES and MMS. As such either side or both may enforce the specified policy.

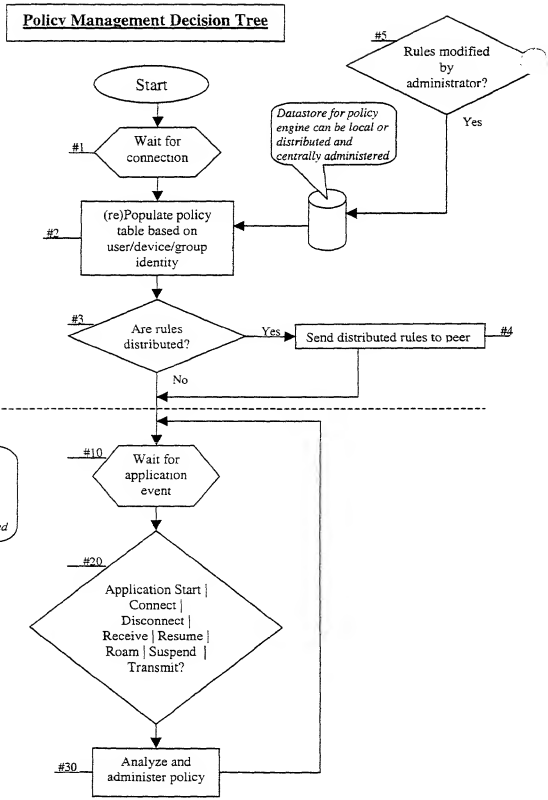

Policy Management Decision Tree
Start
#1
Wait for connection
#2
(re)Populate policy table based on user/device/group identity
#3
Are rules distributed?
Yes
Send distributed rules to peer
#4
No
#5
Rules modified by administrator?
Yes
Datastore for policy engine can be local or distributed and centrally administered
At step #10, duplicate processes can be run on both the MES and MMS. For each application session the rules are analyzed and administered
#10
Wait for application event
#20
Application Start | Connect | Disconnect | Receive | Resume | Roam | Suspend | Transmit?
#30
Analyze and administer policy

FIG. 25